CausalGraph2LLM
Evaluating LLMs for Causal Queries

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Causality is essential in scientific research, enabling researchers to interpret true relationships between variables. These causal relationships are often represented by causal graphs, which are directed acyclic graphs. With the recent advancements in Large Language Models (LLMs), there is an increasing interest in exploring their capabilities in causal reasoning and their potential use to hypothesize causal graphs. These tasks necessitate the LLMs to encode the causal graph effectively for subsequent downstream tasks. In this paper, we propose the first comprehensive benchmark, *CausalGraph2LLM*, encompassing a variety of causal graph settings to assess the causal graph understanding capability of LLMs. We categorize the causal queries into two types: graph-level and node-level queries. We benchmark both open-sourced and closed models for our study. Our findings reveal that while LLMs show promise in this domain, they are highly sensitive to the encoding used. Capable models like GPT-4 and Gemini-1.5 exhibit sensitivity to encoding, with deviations of about 60%. We further demonstrate this sensitivity for downstream causal intervention tasks. Moreover, we observe that LLMs can often display biases when presented with contextual information about a causal graph, potentially stemming from their parametric memory.

1 Introduction

The recent success of Large Language Models (LLMs) [Brown et al., 2020, Achiam et al., 2023, Reid et al., 2024] across various applications has opened new avenues beyond traditional Natural Language Processing (NLP) tasks [Srivastava et al., 2022, Wei et al., 2022]. Trained on massive corpora of structured and unstructured data [Achiam et al., 2023], these models have shown the ability to extract insights and exhibit emergent behaviors that can be harnessed across a wide range of applications [Bubeck et al., 2023, Qi et al., 2023, Wang et al., 2023, Zhao et al., 2024].

Causal reasoning plays a critical role in guiding scientific research to establish causal relationships between variables [Pearl, 2009]. These relationships are often modeled using causal graphs, which are directed and acyclic. Traditionally, causal inference and discovery rely on observational data from experiments [Spirtes and Zhang, 2016, Nogueira et al., 2022, Huang et al., 2020, Cooper and Yoo, 2013]. However, inferring causal graphs from observational data alone is challenging [Spirtes and Zhang, 2016, Brouillard et al., 2020], often necessitating additional domain knowledge, typically from Randomized Controlled Trials. This bottleneck has sparked interest in the potential of LLMs to aid in causal discovery [Vashishtha et al., 2023, Anonymous, 2023, Liu et al., 2024, Ban et al., 2023b,a]. The current paradigm for LLMs in causal discovery usually involves leveraging metadata, particularly variable names, to guide models in identifying and interpreting causal relationships.

Submitted to NeurIPS 2024 Workshop on Causality and Large Models (CaLM). Do not distribute.

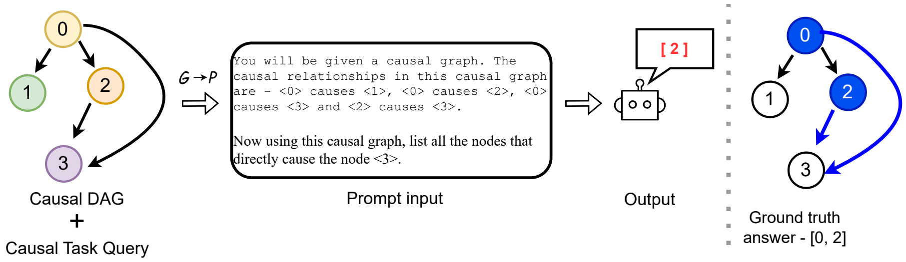

Figure 1: Causal Graphs are ingested into LLMs via prompting strategies which are evaluated for Causal Task Queries.

Existing works utilize LLMs in roles such as priors, critics, and post-processors in causality-related tasks.

Although LLMs have shown competitive performance [Anonymous, 2023] against traditional data-driven methods, their effectiveness is limited by their sequential text-based training paradigm. Current models often require users to decompose causal reasoning tasks into textualizing a causal graph followed by task-specific prompts. Consequently, LLMs must handle and manipulate textual representations of causal graphs efficiently. This assumed capability of processing causal graphs as text with any encoding is often unexamined in current research. Recent works have demonstrated sensitivity to prompts and encoding strategies for graphs [Fatemi et al., 2024a,b], but these are focused on graph theory tasks rather than causal queries.

In this work, we challenge this assumption and evaluate the encoding capabilities of LLMs for causal graphs. By introducing our benchmark, we highlight the strengths and limitations of these models in encoding causal graphs. To maximize LLMs' potential for causality, it is essential to understand their risks and limitations, particularly regarding biases from training data and variable performance based on prompting strategy and task. Proper evaluation and consideration of these aspects are crucial when using LLMs for causal reasoning. Given the application of LLMs as causal hypothesis generators [Liu et al., 2024], it is critical to assess their basic understanding of causal graphs before progressing to complex tasks. Addressing these challenges early can refine models, making them more robust for causal reasoning and hypothesis generation.

In this work, we investigate LLMs' ability to encode causal graphs and assist with causal reasoning tasks. We introduce the first benchmark, *CausalGraph2LLM*, to analyze LLMs in causal graph understanding tasks. We assess various LLMs across a wide spectrum of tasks, inspired by potential subtasks relevant to downstream applications. This benchmark serves as a foundational reference for future research employing LLMs in causal reasoning tasks. Our contributions include:

- We conduct a comprehensive study on techniques for encoding causal graphs into text for LLMs.
- We decompose the task into subtasks involving graph-level and node-level queries to evaluate LLMs' causal reasoning capabilities.
- We explore various graph encoding strategies, drawing from existing literature on causal LLMs and graph theory.
- Our work identifies biases in model performance related to pretraining data context.
- We perform extensive experiments on both open-source and closed models, highlighting the limitations of LLMs in fully understanding causal graphs.

2 Benchmark

Causal graph understanding is crucial for leveraging LLMs in causal graph-based tasks. This benchmark evaluates LLMs' ability to interpret and utilize causal graphs, essential for causal inference and discovery applications. An overview is provided in Figure 1. By assessing how well these models process and understand causal graph structures, we gain insights into their potential and limitations for complex reasoning tasks.

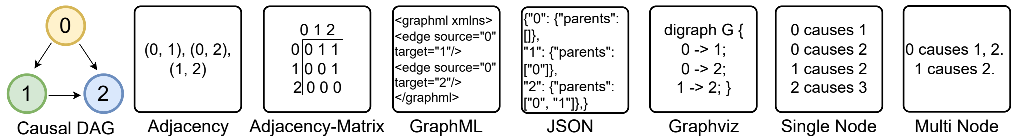

Figure 2: Different prompting transformation functions for the same causal graph.

2.1 Preliminaries

Causal graphs are effective tools for representing variable interactions in research, typically depicted as Directed Acyclic Graphs (DAGs). These graphs help researchers determine which variables to control to reduce bias and identify potential biases that could arise if certain variables are controlled.

A causal graph is defined as $G = (V, E)$, where V is a set of nodes $\{v_1, v_2, \ldots, v_n\}$, each representing a variable, and E is a set of directed edges $\{(v_i, v_j)\}$ indicating causal effects between nodes. The graph is acyclic, implying no causal feedback loops.

Instruction-tuned LLMs are increasingly used to infer causal structures through prompting. We benchmark LLMs' understanding of causal graphs by converting graphs into verbalized prompts using a function $p : G \to P$, where P is the space of all possible prompts. We experiment with seven encoding strategies derived from current literature, as illustrated in Figure 2.

2.2 Tasks

We consider various causality-based tasks that are critical for assessing LLMs' understanding of causal graphs. After encoding the graph into a prompt, a task-specific question prompt is appended to evaluate the LLM's reasoning capabilities. Key tasks include:

- **Child and Parent:** Identifies direct causal effects where one node is a parent of another.
- **Source and Sink:** Identifies nodes without incoming (source) or outgoing (sink) edges, representing starting or ending points in causal chains.
- **Mediator:** Detects nodes that lie on paths between other nodes, mediating causal effects.
- **Confounder:** Identifies nodes influencing two or more other nodes, potentially inducing bias if uncontrolled.

These tasks evaluate an LLM's ability to recognize and interpret causal graph structures from multiple causal reasoning perspectives.

2.3 Experimental Setup

We evaluate the benchmark using diverse datasets, including synthetic, semi-synthetic, and real-world scenarios. Synthetic DAGs are constructed to control graph complexity, and commonly used causal graphs from recent literature [Ban et al., 2023b, Vashishtha et al., 2023, Ban et al., 2023a] are included. For contextual datasets, we use graphs from the BNLearn repository, such as Insurance: $G(27, 52)$ [Binder et al., 1997], and Alarm: $G(37, 46)$ [Beinlich et al., 1989]. We assess the benchmark on a range of models, including GritLM [Muennighoff et al., 2024], GPT-3.5 [Brown et al., 2020], GPT-4 [Achiam et al., 2023], Mistral-7B-Instruct-v0.2 [Jiang et al., 2023], Mixtral-8x7BInstruct-v0.1 [Jiang et al., 2024], and Gemini [Reid et al., 2024][1].

3 Results

In this section, we share our benchmark results on causal graph understanding through causal queries. We investigate how effectively LLMs can interpret and reason about causal graphs encoded in different formats, addressing both graph-level and node-level queries. Additionally, we explore biases introduced by graph contextual information. For brevity, the variances are reported in Appendix D.1.

[1]Experiments were conducted by authors from Google and CISPA Helmholtz Center for Information Security.

3.1 Basic causal graph queries

To evaluate the baseline causal graph understanding task, we prompt the LLMs with causal query tasks resembling those encountered in larger causal reasoning tasks. We measure the performance of these queries using the F1 score.

Model	Enc	Source	Sink	Parent	Child	Mediator	Confounder	*Avg*
GritLM	JSON	0.25	0.30	0.15	0.20	0.10	0.15	$0.19_{\pm 0.10}$
	Adjacency	0.20	0.26	0.12	0.06	0.35	0.26	$0.20_{\pm 0.12}$
	Adjacency-M	0.00	0.05	0.08	0.11	0.06	0.06	$0.06_{\pm 0.03}$
	GraphML	0.38	0.24	0.14	0.21	0.18	0.29	$\underline{0.24_{\pm 0.08}}$
	GraphViz	0.15	0.25	0.19	0.23	0.17	0.22	$0.20_{\pm 0.03}$
	Multi node	0.11	0.32	0.10	0.43	0.19	0.24	$0.23_{\pm 0.12}$
	Single node	0.12	0.34	0.18	0.36	0.25	0.17	$0.23_{\pm 0.10}$
	$\bar{x}$ / σ	0.18 / 0.38	0.27 / 0.29	0.14 / 0.11	0.20 / 0.37	0.19 / 0.29	0.20 / 0.23	
Mistral	JSON	0.30	0.04	0.58	0.20	0.21	0.19	$0.25_{\pm 0.18}$
	Adjacency	0.36	0.15	0.26	0.56	0.28	0.31	$0.32_{\pm 0.13}$
	Adjacency-M	0.07	0.16	0.11	0.10	0.09	0.10	$0.10_{\pm 0.03}$
	GraphML	0.18	0.21	0.31	0.59	0.46	0.61	$\underline{0.39_{\pm 0.18}}$
	GraphViz	0.35	0.27	0.36	0.43	0.46	0.39	$0.37_{\pm 0.06}$
	Multi node	0.37	0.25	0.24	0.45	0.31	0.42	$0.34_{\pm 0.08}$
	Single node	0.50	0.22	0.44	0.43	0.33	0.20	$0.35_{\pm 0.12}$
	$\bar{x}$ / σ	0.32 / 0.43	0.21 / 0.23	0.30 / 0.47	0.38 / 0.49	0.33 / 0.41	0.30 / 0.27	
Mixtral	JSON	0.61	0.04	0.54	0.18	0.22	0.43	$0.33_{\pm 0.22}$
	Adjacency	0.32	0.56	0.45	0.49	0.44	0.32	$0.43_{\pm 0.09}$
	Adjacency-M	0.11	0.08	0.09	0.12	0.10	0.09	$0.10_{\pm 0.01}$
	GraphML	0.38	0.14	0.30	0.39	0.45	0.37	$0.34_{\pm 0.10}$
	GraphViz	0.76	0.50	0.46	0.39	0.55	0.37	$\underline{0.50_{\pm 0.14}}$
	Multi node	0.39	0.49	0.27	0.29	0.49	0.19	$0.35_{\pm 0.12}$
	Single node	0.71	0.33	0.48	0.42	0.54	0.39	$0.48_{\pm 0.13}$
	$\bar{x}$ / σ	0.48 / 0.65	0.31 / 0.52	0.38 / 0.37	0.33 / 0.45	0.44 / 0.34	0.33 / 0.40	
GPT-3.5	JSON	0.75	0.25	0.47	0.08	0.37	0.26	$0.36_{\pm 0.23}$
	Adjacency	0.47	0.29	0.44	0.77	0.65	0.84	$\underline{0.57_{\pm 0.21}}$
	Adjacency-M	0.05	0.19	0.10	0.11	0.15	0.10	$0.12_{\pm 0.11}$
	GraphML	0.72	0.51	0.50	0.61	0.36	0.37	$0.51_{\pm 0.13}$
	GraphViz	0.70	0.18	0.58	0.77	0.55	0.43	$0.53_{\pm 0.12}$
	Multi node	0.39	0.24	0.50	0.70	0.64	0.59	$0.51_{\pm 0.17}$
	Single node	0.70	0.30	0.56	0.67	0.55	0.45	$0.54_{\pm 0.14}$
	$\bar{x}$ / σ	0.57 / 0.70	0.31 / 0.33	0.48 / 0.48	0.50 / 0.69	0.50 / 0.50	0.47 / 0.74	
Gemini	JSON	0.80	0.77	0.97	0.56	0.68	0.72	$\underline{0.76_{\pm 0.13}}$
	Adjacency	0.53	0.62	0.66	0.74	0.64	0.73	$0.66_{\pm 0.07}$
	Adjacency-M	0.12	0.49	0.07	0.12	0.11	0.07	$0.22_{\pm 0.16}$
	GraphML	0.84	0.54	0.76	0.56	0.67	0.60	$0.67_{\pm 0.11}$
	GraphViz	0.48	0.56	0.57	0.64	0.59	0.69	$0.58_{\pm 0.07}$
	Multi node	0.50	0.73	0.70	0.70	0.63	0.59	$0.64_{\pm 0.08}$
	Single node	0.88	0.62	0.69	0.73	0.71	0.57	$0.71_{\pm 0.10}$
	$\bar{x}$ / σ	0.65 / 0.76	0.62 / 0.28	0.69 / 0.90	0.64/0.62	0.64 / 0.66	0.62 / 0.68	
GPT-4	JSON	0.68	0.69	0.52	0.43	0.75	0.74	$0.80_{\pm 0.13}$
	Adjacency	0.77	0.58	0.69	0.69	0.84	0.75	$0.73_{\pm 0.09}$
	Adjacency-M	0.10	0.18	0.21	0.11	0.10	0.13	$0.14_{\pm 0.04}$
	GraphML	0.80	0.80	0.85	0.90	0.76	0.75	$\underline{0.81_{\pm 0.05}}$
	GraphViz	0.67	0.67	0.80	0.85	0.70	0.69	$0.71_{\pm 0.07}$
	Multi node	0.66	0.65	0.73	0.88	0.84	0.79	$0.75_{\pm 0.09}$
	Single node	0.80	0.42	0.89	0.90	0.69	0.87	$0.77_{\pm 0.18}$
	$\bar{x}$ / σ	0.68 / 0.70	0.61 / 0.62	0.71 / 0.68	0.71 / 0.79	0.73 / 0.80	0.72 / 0.74	

Table 1: Performance comparison across methods and encodings. $\bar{x}$ denotes the average performance for each task and σ denotes the difference between the best and the worst encoding.

LLMs struggle with simple causal query tasks. From Table 1, we observe a range of performances across different models and encoding types, highlighting the variability in how well each LLM handles causal graph encoding and interpretation. Out of Source and Sink based queries, interestingly the LLM has stronger performance on performing source tasks. We ablate in Appendix D.2 and observe that the order of causal graph description also has an impact on the performance of source and sink queries. This implies that the model's understanding of causal relationships may be influenced by the sequence in which information is presented. More complex tasks such as identifying mediators seem to be more challenging since the task of identifying a mediator can be intuitively thought of as breaking the task into *child* and *parent* identifications.

Average Performance. Observing the average performance for each model across different encodings suggests that the LLMs are highly sensitive to graph encoding. Adjacency-matrix encoding generally results in the lowest average performance across all models, despite being a popular format to represent causal graphs.

High sensitivity to causal graph representation. We observe that different encodings for the same causal graphs have different performances across each causal query. For instance, for the Mistral model, JSON encoding has the F1 score of 0.21, however for GraphML or GraphViz encoding the performance increases to 0.46 for the Mediator task. GPT-4 and Gemini 1.5 Pro perform exceptionally well with certain encodings like GraphML and JSON, respectively, indicating that these formats might align better with the potential pretraining of the model. GritLM and Mistral show greater variability in their average performance, highlighting their sensitivity to the encoding methods used.

Correlation between Query and Encodings. Some queries may seem easier due to the definition of the encoding and its potential alignment with the encoding. For instance, for JSON encoding, identifying parent nodes might be relatively easier for all LLMs. This could be because the JSON-based prompt used by Abdulaal et al. [2024] defines the dictionary by specifying the parents of each node. This alignment between the query and encoding likely facilitates the model's understanding of the causal relationships, resulting in improved performance on tasks involving parent nodes. This shows the importance of considering the encoding method coupled with the query when concerned with a causal graph based reasoning task.

3.2 Effect of pretraining knowledge on causal graph understanding

Previously, we used synthetic causal graphs to evaluate LLMs' reasoning about causal relationships. Now, we assess the impact of pretraining knowledge on causal graph understanding by testing contextualized causal graphs. This experiment utilizes known causal DAGs, Insurance [Binder et al., 1997] and Alarm [Beinlich et al., 1989], presented in two formats: one with semantically meaningful labels and one with random identifiers.

Enc	**Model**	**Source**		**Sink**		**Parent**		**Child**	
		w/o	w	w/o	w	w/o	w	w/o	w
Insurance	GritLM	0.55	0.72 $_{+0.17}$	0.43	0.65 $_{+0.22}$	0.40	0.62 $_{+0.22}$	0.35	0.53 $_{+0.18}$
	Mistral	0.66	0.74 $_{+0.08}$	0.21	0.43 $_{+0.22}$	0.43	0.65 $_{+0.22}$	0.50	0.69 $_{+0.19}$
	Mixtral	0.66	0.81 $_{+0.15}$	0.32	0.47 $_{+0.15}$	0.36	0.54 $_{+0.18}$	0.49	0.72 $_{+0.23}$
	GPT-3.5	0.48	0.74 $_{+0.26}$	0.40	0.68 $_{+0.28}$	0.39	0.68 $_{+0.29}$	0.42	0.66 $_{+0.24}$
	Gemini	0.72	0.78 $_{+0.06}$	0.65	0.74 $_{+0.09}$	0.57	0.74 $_{+0.17}$	0.73	0.79 $_{+0.06}$
	GPT-4	0.68	0.79 $_{+0.11}$	0.83	0.92 $_{+0.09}$	0.75	0.92 $_{+0.17}$	0.88	0.80 $_{-0.08}$
Alarm	GritLM	0.52	0.59 $_{+0.07}$	0.47	0.54 $_{+0.07}$	0.36	0.54 $_{+0.18}$	0.52	0.61 $_{+0.09}$
	Mistral	0.33	0.81 $_{+0.48}$	0.46	0.63 $_{+0.17}$	0.31	0.54 $_{+0.23}$	0.46	0.58 $_{+0.12}$
	Mixtral	0.66	0.81 $_{+0.15}$	0.32	0.47 $_{+0.15}$	0.36	0.54 $_{+0.18}$	0.49	0.72 $_{+0.23}$
	GPT-3.5	0.60	0.76 $_{+0.16}$	0.69	0.84 $_{+0.15}$	0.38	0.48 $_{+0.10}$	0.42	0.38 $_{-0.04}$
	Gemini	0.77	0.84 $_{+0.07}$	0.68	0.82 $_{+0.14}$	0.71	0.69 $_{-0.02}$	0.49	0.55 $_{+0.06}$
	GPT-4	0.82	0.83 $_{+0.01}$	0.78	0.89 $_{+0.11}$	0.66	0.82 $_{+0.16}$	0.77	0.68 $_{-0.09}$

Table 2: Performance of different models across Alarm and Insurance graphs. w/o - without context w - with contextual variables. The results are averages across the encodings.

Table 6 shows that contextual knowledge improves performance across models, leveraging LLMs' pretraining on vast text corpora. Semantically meaningful labels aid in more accurate causal interpretations by activating the model's parametric memory.

Risks of Contextual Knowledge Dependence. The reliance on contextual knowledge, while beneficial, introduces risks such as biases from the language and cultural context of training data. For example, GPT-4's increased false positives in the Child query for the Insurance graph suggest over-reliance on pretraining priors, aligning with findings from [Vashishtha et al., 2023]. Performance also drops with anti-commonsense DAGs, highlighting the potential for errors when causal directions deviate from LLM pretraining biases.

3.3 Node-based Queries Simplify LLM Performance

In our previous experiments, we focused on *graph overview* tasks, which required LLMs to identify all instances of specific node types (e.g., source, sink) within a causal graph, demanding a comprehensive understanding of the entire graph structure. To simplify this, we decompose these tasks into binary *node-inspection* queries, where the LLM evaluates whether a given node fits a specified type.

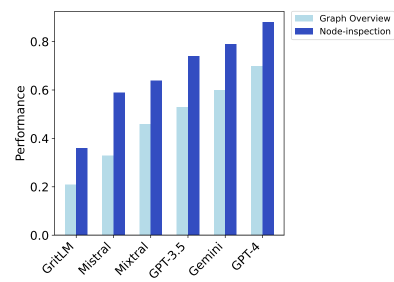

Figure 3: Performance: Node inspection vs. graph overview.

This breakdown reduces the processing complexity, allowing LLMs to focus on individual nodes rather than the entire graph. As shown in Figure 3, LLMs perform better on *node-inspection* tasks due to the localized nature of the queries. The lower performance on *graph overview* tasks is likely due to the need for holistic graph comprehension and the potential cascading effect of errors in node identification. In contrast, *node-inspection* tasks minimize the impact of individual errors, leading to improved accuracy.

3.3.1 Overestimation and Underestimation Biases

For binary *node-inspection* tasks, we analyze false positives (FP) and false negatives (FN) for each LLM, providing insight into the nature of their errors. False positives occur when a model incorrectly identifies a node type, while false negatives occur when it fails to identify a correct type. We compute the ratio (τ) of FP to FN, averaged across all tasks, where $\tau > 1$ indicates a bias towards overestimation (more FPs), and $\tau < 1$ indicates a bias towards underestimation (more FNs).

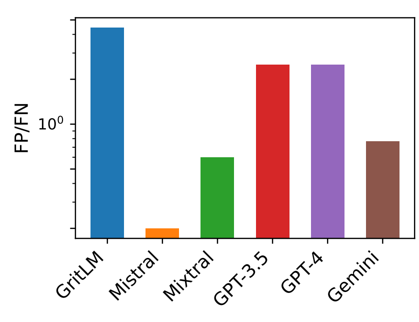

Figure 4: Evaluation of over- and underestimation biases.

Figure 4 shows that GritLM, GPT-3.5, and GPT-4 have $\tau > 1$, suggesting a tendency towards overestimation, even without contextual influences, aligning with recent findings [Herrera-Berg et al., 2023, Li et al., 2024]. Conversely, Gemini, Mistral, and Mixtral exhibit higher FN rates, indicating underestimation, potentially influenced by RLHF fine-tuning stages. Further investigation is needed to explore these biases in causal queries.

4 Conclusion

With the increasing of LLMs to assist with causal inference and causal discovery tasks, it is important to explore the opportunities and the limitations due to the nature of LLMs. In this paper, we proposed the first benchmark, *CausalGraph2LLM* to evaluate the encoding capabilities of LLMs for causal DAGs, encompassing both graph-level and node-level queries. Our findings also shed light on the potential risks associated with employing LLMs for causal reasoning tasks, particularly emphasizing the potential biases stemming from their pre-trained knowledge. These insights serve as a valuable reference for future research leveraging LLMs in causal DAG manipulation.

References

Ahmed Abdulaal, adamos hadjivasiliou, Nina Montana-Brown, Tiantian He, Ayodeji Ijishakin, Ivana Drobnjak, Daniel C. Castro, and Daniel C. Alexander. Causal modelling agents: Causal graph discovery through synergising metadata- and data-driven reasoning. In *ICLR*, 2024. URL `https://openreview.net/forum?id=pAoqRlTBtY`.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv*, 2023.

Microsoft Research AI4Science and Microsoft Azure Quantum. The impact of large language models on scientific discovery: a preliminary study using gpt-4. *arXiv*, 2023.

Anonymous. Causal modelling agents: Causal graph discovery through synergising metadata- and data-driven reasoning. In *ICLR*, 2023. URL `https://openreview.net/forum?id=pAoqRlTBtY`. under review.

Taiyu Ban, Lyuzhou Chen, Derui Lyu, Xiangyu Wang, and Huanhuan Chen. Causal structure learning supervised by large language model. *arXiv*, 2023a.

Taiyu Ban, Lyvzhou Chen, Xiangyu Wang, and Huanhuan Chen. From query tools to causal architects: Harnessing large language models for advanced causal discovery from data. *arXiv*, 2023b.

Ingo A Beinlich, Henri Jacques Suermondt, R Martin Chavez, and Gregory F Cooper. The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks. In *AIME 89: Second European Conference on Artificial Intelligence in Medicine, London, August 29th–31st 1989. Proceedings*, pages 247–256. Springer, 1989.

John Binder, Daphne Koller, Stuart Russell, and Keiji Kanazawa. Adaptive probabilistic networks with hidden variables. *Machine Learning*, 29:213–244, 1997.

Philippe Brouillard, Sébastien Lachapelle, Alexandre Lacoste, Simon Lacoste-Julien, and Alexandre Drouin. Differentiable causal discovery from interventional data. *Advances in Neural Information Processing Systems*, 33:21865–21877, 2020.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv*, 2023.

Hengrui Cai, Shengjie Liu, and Rui Song. Is knowledge all large language models needed for causal reasoning? *arXiv*, 2023.

Anthony C Constantinou, Zhigao Guo, and Neville K Kitson. The impact of prior knowledge on causal structure learning. *Knowledge and Information Systems*, 65(8):3385–3434, 2023.

Gregory F Cooper and Changwon Yoo. Causal discovery from a mixture of experimental and observational data. *arXiv*, 2013.

Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *arXiv*, 2023.

Dorottya Demszky, Diyi Yang, David S Yeager, Christopher J Bryan, Margarett Clapper, Susannah Chandhok, Johannes C Eichstaedt, Cameron Hecht, Jeremy Jamieson, Meghann Johnson, et al. Using large language models in psychology. *Nature Reviews Psychology*, 2(11):688–701, 2023.

Bahare Fatemi, Jonathan Halcrow, and Bryan Perozzi. Talk like a graph: Encoding graphs for large language models. In *ICLR*, 2024a.

Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. Test of time: A benchmark for evaluating llms on temporal reasoning. *arXiv preprint arXiv:2406.09170*, 2024b.

Roxana Girju, Dan I Moldovan, et al. Text mining for causal relations. In *FLAIRS conference*, pages 360–364, 2002.

Oktie Hassanzadeh, Debarun Bhattacharjya, Mark Feblowitz, Kavitha Srinivas, Michael Perrone, Shirin Sohrabi, and Michael Katz. Causal knowledge extraction through large-scale text mining. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 13610–13611, 2020.

Eugenio Herrera-Berg, Tomás Vergara Browne, Pablo León-Villagrá, Marc-Lluís Vives, and Cristian Buc Calderon. Large language models are biased to overestimate profoundness. In *EMNLP*, 2023. URL `https://openreview.net/forum?id=PT63nNpyKg`.

Biwei Huang, Kun Zhang, Jiji Zhang, Joseph Ramsey, Ruben Sanchez-Romero, Clark Glymour, and Bernhard Schölkopf. Causal discovery from heterogeneous/nonstationary data. *Journal of Machine Learning Research*, 21(89):1–53, 2020.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv*, 2023.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv*, 2024.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. CLadder: Assessing causal reasoning in language models. In *NeurIPS*, 2023a.

Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona Diab, and Bernhard Schölkopf. Can large language models infer causation from correlation? *arXiv*, 2023b.

Thomas Jiralerspong, Xiaoyin Chen, Yash More, Vedant Shah, and Yoshua Bengio. Efficient causal graph discovery using large language models. *arXiv preprint arXiv:2402.01207*, 2024.

Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language models: Opening a new frontier for causality. *arXiv*, 2023.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022.

Tianlin Li, Xiaoyu Zhang, Chao Du, Tianyu Pang, Qian Liu, Qing Guo, Chao Shen, and Yang Liu. Your large language model is secretly a fairness proponent and you should prompt it like one. *arXiv*, 2024.

Chenxi Liu, Yongqiang Chen, Tongliang Liu, Mingming Gong, James Cheng, Bo Han, and Kun Zhang. Discovery of the hidden world with large language models. *arXiv*, 2024.

Stephanie Long, Tibor Schuster, Alexandre Piché, ServiceNow Research, et al. Can large language models build causal graphs? *arXiv*, 2023.

Niklas Muennighoff, Hongjin Su, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. Generative representational instruction tuning. *arXiv*, 2024.

Ana Rita Nogueira, Andrea Pugnana, Salvatore Ruggieri, Dino Pedreschi, and João Gama. Methods and tools for causal discovery and causal inference. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, 12(2):e1449, 2022.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Biqing Qi, Kaiyan Zhang, Haoxiang Li, Kai Tian, Sihang Zeng, Zhang-Ren Chen, and Bowen Zhou. Large language models are zero shot hypothesis proposers. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv*, 2024.

Peter Spirtes and Kun Zhang. Causal discovery and inference: concepts and recent methodological advances. In *Applied informatics*, volume 3, pages 1–28. Springer, 2016.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2001.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv*, 2022.

Fiona Anting Tan, Xinyu Zuo, and See-Kiong Ng. Unicausal: Unified benchmark and repository for causal text mining. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 248–262. Springer, 2023.

Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N Balasubramanian, and Amit Sharma. Causal inference using llm-guided discovery. *arXiv*, 2023.

Ruocheng Wang, Eric Zelikman, Gabriel Poesia, Yewen Pu, Nick Haber, and Noah D Goodman. Hypothesis search: Inductive reasoning with language models. *arXiv*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *NeurIPS*, 35: 24824–24837, 2022.

Matej Zečević, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. Causal parrots: Large language models may talk causality but are not causal. *TMLR*, 2023.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642, 2024.

A Limitations and Future Work

The scope of the evaluation is primarily limited to synthetic and well-known causal graphs, which may not fully capture the complexity of real-world causal graphs. We presented 6 diverse tasks, which can be built upon for future work. Future work can expand the diversity of causal graphs and models evaluated, develop more robust encoding techniques, and explore methods to mitigate contextual biases. Enhancing the models' ability to handle complex tasks and improving downstream task performance will also be crucial. Additionally, a deeper investigation into bias sources can provide a more nuanced understanding of LLM capabilities in causal inference. Given the modular nature of the benchmark, we aim to continue to build up this benchmark to assess newer models as they come.

B Reproduciblility

We will release our code, prompts, evaluation setup, and all models' outputs of our experiments. For reproducibility, we used temperature 0 and top-p value as 1 across all of the models. We also mentioned the snapshot of the model used.

The Alarm and Insurance datasets are under CC BY-SA 3.0 which allows us to freely modify the datasets for benchmarking. Our benchmark will be released under the CC BY-SA License.

For Mistral, Mixtral and GritLM models, were run via Helmholtz Jeulich. Mistral and GritLM were run on 1 A100 GPU whereas Mixtral was run on 8 A100 GPUs. Since we used off-the shelf LLM, each graph-level experiment took no more than 30 minutes to run (longer for mediator, child, parent, confounder whereas source and sink took ≈ 3 mins to run). Since the models were run by Jeulich API, it is difficult to calculate the entire compute, however all of the experiments for each model took ≈ 38 hours. GPT-3.5 GPT-4 were accessed via API.

B.1 Dataset descriptions

The datasets used can be divided into two: 1. realistic datasets and 2. synthetic datasets.

We use the two real-world-based datasets. These are semi-synthetic datasets available from the BNLearn library. The first graph, named **Alarm**, is a well-known benchmark in the field of causal inference. The Alarm dataset (see Figure 11) is designed to model the relationships and dependencies in an intensive care unit (ICU) monitoring system. It includes variables such as heart rate, blood pressure, and other vital signs, making it a complex and realistic representation of medical data. This dataset is particularly useful for evaluating the ability of LLMs to handle intricate causal relationships in a medical high-stakes environment.

The second dataset, **Insurance**, is another widely used benchmark that models the risk factors and dependencies in the insurance domain. This graph (see Figure 12) includes variables related to policyholders, such as age, driving history, and vehicle type, and their relationships to insurance claims and premiums. The Insurance dataset provides a different context from the medical domain, allowing us to assess the versatility of LLMs in understanding and reasoning about causal relationships in a financial setting.

B.2 Synthetic dataset

In addition to real-world-based datasets, we created synthetic datasets with varying levels of difficulty to rigorously evaluate the performance of LLMs. These synthetic datasets were designed to systematically vary in complexity by adjusting the number of nodes and edges in the causal graphs. This variation allows us to assess how well the models handle different levels of graph complexity and density. The synthetic datasets serve as a controlled environment to test the models' ability to interpret and reason about causal relationships under varying conditions. By incrementally increasing the number of edges while keeping the number of nodes constant, we can observe how the models' performance scales with the complexity of the causal structure. This approach provides valuable insights into the strengths and limitations of LLMs in handling more intricate causal graphs, which is crucial for understanding their potential applications in real-world scenarios. For the experiments, we synthesized graphs with 20 and 30 nodes. For each of these node variables, we experimented with different densities of nodes. Hence we had density = 1 x nodes, 1.5 x nodes and 2 x nodes.

C Prompting stratergies

Adjacency

(0 , 1)
(0 , 2)
(1 , 3)
(2 , 3)
(2 , 4)
(3 , 4)
(0 , 3)
(1 , 4)
(0 , 4)
(1 , 2)

Adjacency Matrix

	0	1	2	3	4
0	0	1	1	1	1
1	0	0	1	1	1
2	0	0	0	1	1
3	0	0	0	0	1
4	0	0	0	0	0

GraphML

```
<?xml version="1.0" encoding="UTF-8"?>
<graphml xmlns="http://graphml.graphdrawing.org/xmlns">
  <graph edgedefault="directed">
    <node id="0"/>
    <node id="1"/>
    <node id="2"/>
    <node id="3"/>
    <node id="4"/>
    <edge source="0" target="1"/>
    <edge source="0" target="2"/>
    <edge source="1" target="3"/>
    <edge source="2" target="3"/>
    <edge source="2" target="4"/>
    <edge source="3" target="4"/>
    <edge source="0" target="3"/>
    <edge source="1" target="4"/>
    <edge source="0" target="4"/>
    <edge source="1" target="2"/>
  </graph>
</graphml>
```

GraphViz

```
digraph G {
  0 -> 1;
  0 -> 2;
  1 -> 3;
  2 -> 3;
  2 -> 4;
  3 -> 4;
  0 -> 3;
  1 -> 4;
  0 -> 4;
  1 -> 2;
}
```

373

JSON

```
{
    "0": {
        "parents": []
    },
    "1": {
        "parents": [
            "0"
        ]
    },
    "2": {
        "parents": [
            "0",
            "1"
        ]
    },
    "3": {
        "parents": [
            "0",
            "1",
            "2"
        ]
    },
    "4": {
        "parents": [
            "0",
            "1",
            "2",
            "3"
        ]
    }
}
```

374

Multi node

375 0 causes 1, 2, 3, 4. 1 causes 3, 4, 2. 2 causes 3, 4. 3 causes 4.

Single node

0 causes 1. 0 causes 2. 0 causes 3. 0 causes 4. 1 causes 3. 1 causes 4. 1 causes 2. 2 causes 3. 2 causes 4. 3 causes 4.

D Experiments

D.1 Variance

Model	Enc	**Source**	**Sink**	**Parent**	**Child**	**Mediator**	**Confounder**	*Avg*
GritLM	JSON	0.25 ±0.07	0.30 ±0.05	0.15 ±0.02	0.20 ±0.07	0.10 ±0.08	0.15 ±0.07	0.19±0.10
	Adjacency	0.20 ±0.03	0.26 ±0.04	0.12 ±0.01	0.06 ±0.02	0.35 ±0.05	0.26 ±0.04	0.20±0.12
	Adjacency-M	0.00 ±0.00	0.05 ±0.01	0.08 ±0.01	0.11 ±0.02	0.06 ±0.01	0.06 ±0.01	0.06±0.03
	GraphML	0.38 ±0.06	0.24 ±0.04	0.14 ±0.03	0.21 ±0.05	0.18 ±0.04	0.29 ±0.05	0.24±0.08
	GraphViz	0.15 ±0.03	0.25 ±0.05	0.19 ±0.04	0.23 ±0.04	0.17 ±0.03	0.22 ±0.04	0.20±0.03
	Multi node	0.11 ±0.02	0.32 ±0.06	0.10 ±0.02	0.43 ±0.08	0.19 ±0.04	0.24 ±0.05	0.23±0.12
	Single node	0.12 ±0.03	0.34 ±0.06	0.18 ±0.04	0.36 ±0.07	0.25 ±0.05	0.17 ±0.04	0.23±0.10
Mistral	JSON	0.30 ±0.03	0.04 ±0.01	0.58 ±0.06	0.20 ±0.02	0.21 ±0.02	0.19 ±0.02	0.25±0.18
	Adjacency	0.36 ±0.04	0.15 ±0.02	0.26 ±0.03	0.56 ±0.06	0.28 ±0.03	0.31 ±0.03	0.32±0.13
	Adjacency-M	0.07 ±0.01	0.16 ±0.02	0.11 ±0.01	0.10 ±0.01	0.09 ±0.01	0.10 ±0.01	0.10±0.03
	GraphML	0.18 ±0.02	0.21 ±0.02	0.31 ±0.03	0.59 ±0.06	0.46 ±0.05	0.61 ±0.06	0.39±0.18
	GraphViz	0.35 ±0.04	0.27 ±0.03	0.36 ±0.04	0.43 ±0.04	0.46 ±0.05	0.39 ±0.04	0.37±0.06
	Multi node	0.37 ±0.04	0.25 ±0.03	0.24 ±0.02	0.45 ±0.05	0.31 ±0.03	0.42 ±0.04	0.34±0.08
	Single node	0.50 ±0.05	0.22 ±0.02	0.44 ±0.04	0.43 ±0.04	0.33 ±0.03	0.20 ±0.02	0.35±0.12
Mixtral	JSON	0.61 ±0.06	0.04 ±0.01	0.54 ±0.05	0.18 ±0.02	0.22 ±0.02	0.43 ±0.04	0.33±0.22
	Adjacency	0.32 ±0.03	0.56 ±0.05	0.45 ±0.04	0.49 ±0.05	0.44 ±0.04	0.32 ±0.03	0.43±0.09
	Adjacency-M	0.11 ±0.01	0.08 ±0.01	0.09 ±0.01	0.12 ±0.01	0.10 ±0.01	0.09 ±0.01	0.10±0.01
	GraphML	0.38 ±0.04	0.14 ±0.01	0.30 ±0.03	0.39 ±0.04	0.45 ±0.04	0.37 ±0.04	0.34±0.10
	GraphViz	0.76 ±0.07	0.50 ±0.05	0.46 ±0.04	0.39 ±0.04	0.55 ±0.05	0.37 ±0.04	0.50±0.14
	Multi node	0.39 ±0.04	0.49 ±0.05	0.27 ±0.03	0.29 ±0.03	0.49 ±0.05	0.19 ±0.02	0.35±0.12
	Single node	0.71 ±0.07	0.33 ±0.03	0.48 ±0.05	0.42 ±0.04	0.54 ±0.05	0.39 ±0.04	0.48±0.13
GPT-3.5	JSON	0.75 ±0.07	0.25 ±0.03	0.47 ±0.05	0.08 ±0.01	0.37 ±0.04	0.26 ±0.03	0.36±0.23
	Adjacency	0.47 ±0.05	0.29 ±0.03	0.44 ±0.04	0.77 ±0.08	0.65 ±0.07	0.84 ±0.09	0.57±0.21
	Adjacency-M	0.05 ±0.01	0.19 ±0.02	0.10 ±0.01	0.11 ±0.01	0.15 ±0.02	0.10 ±0.01	0.12±0.11
	GraphML	0.72 ±0.07	0.51 ±0.05	0.50 ±0.05	0.61 ±0.06	0.36 ±0.04	0.37 ±0.04	0.51±0.13
	GraphViz	0.70 ±0.07	0.18 ±0.02	0.58 ±0.06	0.77 ±0.08	0.55 ±0.06	0.43 ±0.04	0.53±0.12
	Multi node	0.39 ±0.04	0.24 ±0.02	0.50 ±0.05	0.70 ±0.07	0.64 ±0.06	0.59 ±0.06	0.51±0.17
	Single node	0.70 ±0.07	0.30 ±0.03	0.56 ±0.06	0.67 ±0.07	0.55 ±0.06	0.45 ±0.05	0.54±0.14
Gemini	JSON	0.80 ±0.08	0.77 ±0.08	0.97 ±0.10	0.56 ±0.06	0.68 ±0.07	0.72 ±0.07	0.76±0.13
	Adjacency	0.53 ±0.05	0.62 ±0.06	0.66 ±0.07	0.74 ±0.07	0.64 ±0.06	0.73 ±0.07	0.66±0.07
	Adjacency-M	0.12 ±0.01	0.49 ±0.05	0.07 ±0.01	0.12 ±0.01	0.11 ±0.01	0.07 ±0.01	0.22±0.16
	GraphML	0.84 ±0.08	0.54 ±0.05	0.76 ±0.08	0.56 ±0.06	0.67 ±0.07	0.60 ±0.06	0.67±0.11
	GraphViz	0.48 ±0.05	0.56 ±0.06	0.57 ±0.06	0.64 ±0.06	0.59 ±0.06	0.69 ±0.07	0.58±0.07
	Multi node	0.50 ±0.05	0.73 ±0.07	0.70 ±0.07	0.70 ±0.07	0.63 ±0.06	0.59 ±0.06	0.64±0.08
	Single node	0.88 ±0.09	0.62 ±0.06	0.69 ±0.07	0.73 ±0.07	0.71 ±0.07	0.57 ±0.06	0.71±0.10
GPT-4	JSON	0.68 ±0.07	0.69 ±0.07	0.52 ±0.05	0.43 ±0.04	0.75 ±0.08	0.74 ±0.07	0.80±0.13
	Adjacency	0.77 ±0.08	0.58 ±0.06	0.69 ±0.07	0.69 ±0.07	0.84 ±0.08	0.75 ±0.08	0.73±0.09
	Adjacency-M	0.10 ±0.01	0.18 ±0.02	0.21 ±0.02	0.11 ±0.01	0.10 ±0.01	0.13 ±0.01	0.14±0.04
	GraphML	0.80 ±0.08	0.80 ±0.08	0.85 ±0.09	0.90 ±0.09	0.76 ±0.08	0.75 ±0.08	0.81±0.05
	GraphViz	0.67 ±0.07	0.67 ±0.07	0.80 ±0.08	0.85 ±0.09	0.70 ±0.07	0.69 ±0.07	0.71±0.07
	Multi node	0.66 ±0.07	0.65 ±0.07	0.73 ±0.07	0.88 ±0.09	0.84 ±0.08	0.79 ±0.08	0.75±0.09
	Single node	0.80 ±0.08	0.42 ±0.04	0.89 ±0.09	0.90 ±0.09	0.69 ±0.07	0.87 ±0.09	0.77±0.18

Table 3: Performance comparison across methods and encodings.

Enc	Model	Source		Sink		Parent		Child	
		w/o	w	w/o	w	w/o	w	w/o	w
Insurance	GritLM	0.55 ±0.07	0.72 ±0.07	0.43 ±0.03	0.65 ±0.04	0.40 ±0.04	0.62 ±0.05	0.35 ±0.03	0.53 ±0.05
	Mistral	0.66 ±0.06	0.74 ±0.08	0.21 ±0.02	0.43 ±0.09	0.43 ±0.04	0.65 ±0.02	0.50 ±0.05	0.69 ±0.10
	Mixtral	0.66 ±0.06	0.81 ±0.04	0.32 ±0.03	0.47 ±0.05	0.36 ±0.04	0.54 ±0.08	0.49 ±0.05	0.72 ±0.03
	GPT-3.5	0.48 ±0.05	0.74 ±0.05	0.40 ±0.04	0.68 ±0.08	0.39 ±0.04	0.68 ±0.09	0.42 ±0.04	0.66 ±0.04
	Gemini	0.72 ±0.07	0.78 ±0.06	0.65 ±0.06	0.74 ±0.03	0.57 ±0.06	0.74 ±0.07	0.73 ±0.07	0.79 ±0.05
	GPT-4	0.68 ±0.07	0.79 ±0.03	0.83 ±0.08	0.92 ±0.02	0.75 ±0.07	0.92 ±0.06	0.88 ±0.09	0.80 ±0.03
Alarm	GritLM	0.52 ±0.05	0.59 ±0.07	0.47 ±0.05	0.54 ±0.07	0.36 ±0.04	0.54 ±0.04	0.52 ±0.05	0.61 ±0.09
	Mistral	0.33 ±0.03	0.81 ±0.05	0.46 ±0.05	0.63 ±0.04	0.31 ±0.03	0.54 ±0.02	0.46 ±0.05	0.58 ±0.08
	Mixtral	0.66 ±0.06	0.81 ±0.05	0.32 ±0.07	0.47 ±0.03	0.36 ±0.06	0.54 ±0.07	0.49 ±0.05	0.72 ±0.04
	GPT-3.5	0.60 ±0.06	0.76 ±0.09	0.69 ±0.07	0.84 ±0.03	0.38 ±0.04	0.48 ±0.09	0.42 ±0.04	0.38 ±0.04
	Gemini	0.77 ±0.08	0.84 ±0.07	0.68 ±0.07	0.82 ±0.14	0.71 ±0.03	0.69 ±0.02	0.49 ±0.05	0.55 ±0.06
	GPT-4	0.82 ±0.08	0.83 ±0.01	0.78 ±0.08	0.89 ±0.04	0.66 ±0.07	0.82 ±0.06	0.77 ±0.08	0.68 ±0.03

Table 4: Performance of different models across Alarm and Insurance graphs. w/o - without context w - with contextual variables. The results are averages across the encodings.

	JSON	Adjacency	Adjacency-M	GraphML	GraphViz	Multi node	Single node
GritLM	0.44 ±0.04	0.50 ±0.05	0.54 ±0.05	0.53 ±0.05	0.53 ±0.05	0.58 ±0.06	0.54 ±0.05
Mistral	0.43 ±0.04	0.47 ±0.05	0.51 ±0.05	0.50 ±0.05	0.55 ±0.05	0.53 ±0.05	0.58 ±0.06
Mixtral	0.48 ±0.05	0.56 ±0.06	0.51 ±0.05	0.63 ±0.06	0.72 ±0.07	0.61 ±0.06	0.58 ±0.06
GPT-3.5	0.50 ±0.05	0.37 ±0.04	0.48 ±0.05	0.52 ±0.05	0.64 ±0.06	0.56 ±0.06	0.58 ±0.06
Gemini	0.80 ±0.08	0.78 ±0.08	0.54 ±0.05	0.76 ±0.07	0.88 ±0.04	0.78 ±0.08	0.63 ±0.06
GPT-4	0.74 ±0.07	0.74 ±0.07	0.52 ±0.05	0.78 ±0.08	0.88 ±0.03	0.82 ±0.04	0.77 ±0.08

Table 5: Sensitivity to encoding for downstream intervention analysis.

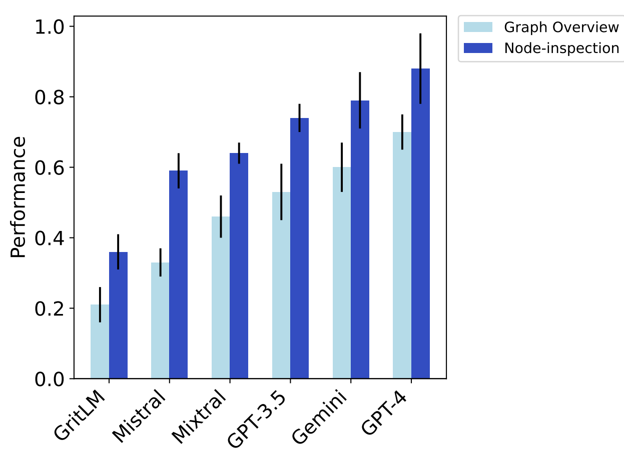

Figure 5: Node inspection vs. graph overview query performances.

D.2 Ordering of prompt matter for causal queries

In BFS, the traversal starts from the source nodes, while in BFS-R, the traversal begins from the sink nodes. The values in the table represent the performance of the models on the tasks, with higher values indicating better performance.

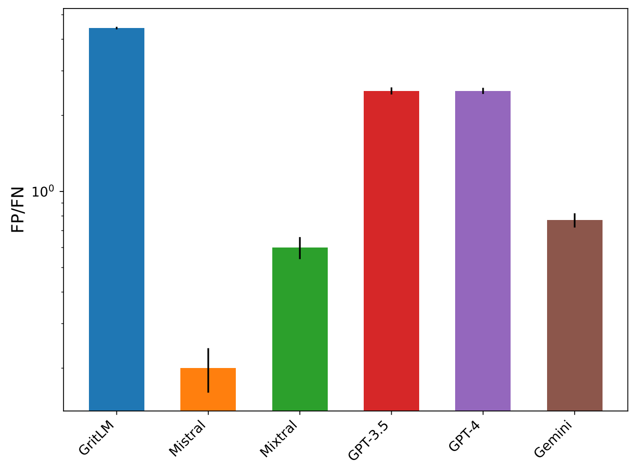

Figure 6: Evaluation of over- and underestimation biases.

The results show that the traversal order significantly impacts the performance of the models. For instance, GritLM performs better on source tasks when the traversal is in BFS order, while it performs better on sink tasks when the traversal is in BFS-R order. This pattern is consistent across all models, suggesting that BFS is more suitable for identifying source nodes, while BFS-R is more suitable for identifying sink nodes.

D	**Model**	**Source**		**Sink**	
		BFS	BFS-R	BFS	BFS-R
Synthetic	GritLM	0.18	0.24	0.27	0.0.47
	Mistral	0.32	0.26	0.21	0.39
	Mixtral	0.48	0.40	0.31	0.44
	GPT-3.5	0.57	0.48	0.31	0.64
	Gemini	0.65	0.54	0.62	0.82
	GPT-4	0.68	0.57	0.61	0.89

Table 6: Comparing the order for prompts, BFS means it starts from source and BFS-R means it starts from sinks.

D.3 Downstream performance under/over bias

In the main paper, we analyzed over and underestimation bias for the binary node inspection task. We can conduct a similar analysis on the downstream task. Here, we observe a similar trend to the estimation biases in Section 4.3.1. Notably, GPT-3.5 and GPT-4 usually have FP/FN ratios closer to 1.

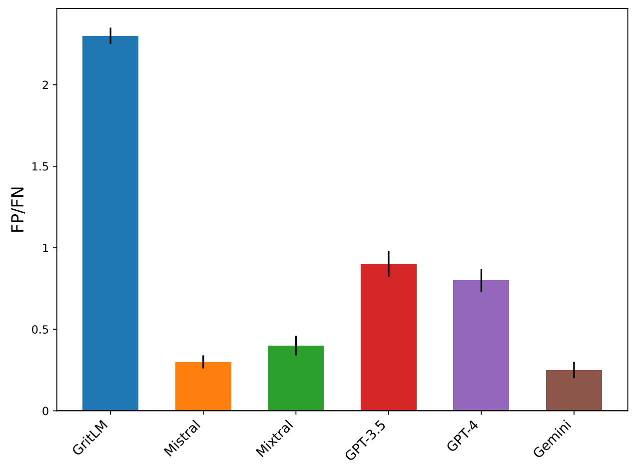

Figure 7: Evaluation of over- and underestimation biases for downstream task.

D.4 Effect of node explanations

In our experimental setup, we took an approach to defining each task for every metric. This was primarily due to the varying terminologies used in causal inference across different academic circles. For instance, what some researchers might refer to as a 'source', others might call a 'root'. To avoid any potential confusion, we provided clear definitions for each term used in our causal queries.

Since pretraining for each model was not known, this adds an element of uncertainty to the task. To counteract this, we explicitly mentioned the query in our experiments. We conducted a set of preliminary experiments without an explanation of the query to demonstrate its effectiveness. The results showed a decrease in model performance, suggesting that providing explicit direction in the form of a mentioned query can be beneficial.

Model	Enc	**Source**	**Sink**	**Parent**	**Child**	**Mediator**	**Confounder**
GPT-3.5	JSON	0.52	0.25	0.47	0.08	0.30	0.31
	Adjacency	0.32	0.26	0.44	0.65	0.72	0.51
	Adjacency-M	0.06	0.15	0.10	0.11	0.08	0.12
	GraphML	0.34	0.38	0.50	0.61	0.37	0.39
	GraphViz	0.42	0.19	0.58	0.77	0.52	0.28
	Multi node	0.39	0.24	0.50	0.70	0.64	0.27
	Single node	0.45	0.27	0.56	0.67	0.39	0.50

Table 7: Performance comparison across methods and encodings for GPT-3.5 without causal query explanations.

D.5 Node Complexity

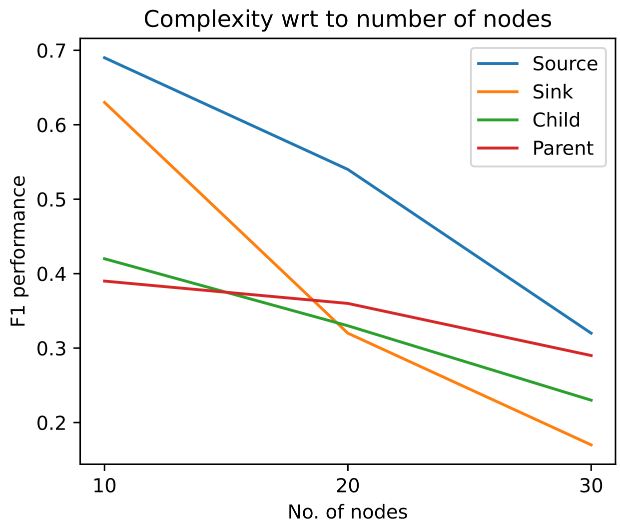

Figure 8: With an increase in graph complexity by increasing the number of nodes, we observe poorer performance of the LLM - Mistral model.

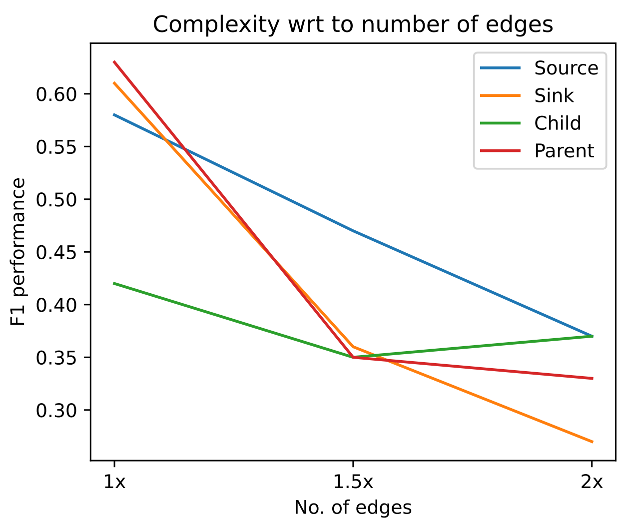

Figure 9: With an increase in graph complexity by increasing the number of edges, we observe poorer performance of the LLM - Mistral model.

D.6 Further models

D.6.1 LLama3.1 models

We additionally tested models from the LLama-3.1 family. We observe that LLama3.1 models also observe sensitivity to the graph encoding.

Model	Enc	Source	Sink	Parent	Child	Mediator	Confounder
8b	JSON	0.30	0.35	0.22	0.25	0.20	0.22
	Adjacency	0.28	0.30	0.18	0.20	0.22	0.23
	GraphML	0.35	0.31	0.28	0.29	0.18	0.31
	Single node	0.36	0.27	0.33	0.40	0.36	0.38
70b	JSON	0.62	0.65	0.52	0.55	0.48	0.50
	Adjacency	0.55	0.58	0.48	0.50	0.50	0.52
	GraphML	0.63	0.62	0.60	0.62	0.58	0.62
	Single node	0.71	0.75	0.72	0.74	0.68	0.69
405b	JSON	0.80	0.82	0.74	0.76	0.70	0.72
	Adjacency	0.75	0.78	0.70	0.72	0.68	0.70
	GraphML	0.85	0.83	0.80	0.82	0.77	0.79
	Single node	0.88	0.90	0.85	0.87	0.82	0.84

Table 8: Performance comparison across methods and encodings. $\bar{x}$ denotes the average performance for each task and σ denotes the difference between the best and the worst encoding.

D.6.2 Multimodal models

In this work we focus on textual encodings into LLMs, however with the developments of multimodal models, we can test LLM's ability to answer causal queries when presented with image inputs. We performed our experiment on GPT-4 model with T=0. Future works can be built upon to test better image inputs for multimodal models.

Source	Sink	Child	Parent	Mediator	Confounder
0.58	0.62	0.71	0.65	0.58	0.63

D.7 Effect of finetuning

In this paper, we focused on zero-shot prompting as the current models have billions of trainable parameters and have been trained on a plethora of training data, potentially including causal graphs. We hence aimed to evaluate how this reflects in the causal queries. Additionally, most current methods utilize LLMs without fine-tuning for causal discovery queries, and our study aimed to replicate this environment to provide a realistic benchmark. We performed QLORA on Mistral 7b specifically on synthetic datasets. As expected, we observed an increase in the performance with finetuning.

	Source	Sink	Parent	Child	Mediator	Confounder
JSON	0.30	0.04	0.58	0.20	0.21	0.19
JSON -FT	0.63	0.36	0.73	0.42	0.33	0.44
GraphML	0.18	0.21	0.31	0.59	0.46	0.61
GraphML - FT	0.47	0.42	0.55	0.73	0.68	0.73

Table 9: Effect of finetuning Mistral 7b model for JSON and GraphML encoding.

E Causal Query explanation

Source

A source node in a causal graph is a variable that does not have any incoming edges, meaning it is not caused by any other variable in the graph.

Sink

A sink node in a causal graph is a variable that does not have any children in the graph, meaning it is not caused by any other variables in the system.

Direct Mediator

A direct mediator in a causal graph is a variable that lies on the direct path between two other immediate variables. Only consider mediators that exist in the direct causal path (not mediated via other mediators).

Confounder

A confounder in a causal graph is a variable that influences both the cause and the effect variables. It is a common cause for both the dependent and independent variables.

Parents

What nodes are the direct causes of Node X?

Child

What nodes are directly caused by Node X?

E.1 Prompt

For further prompt templates, please check the codebase.

Graph level query prompt

Hello. You will be given a causal graph. The causal relationships in this causal graph are - [causal-graph-based-encoding]. Now answer using this causal graph only, name all of the [node-type] in the graph. [node-type-description]. Think step by step. Give reasoning and then give answer within <Answer> [a1,a2,a3..] </Answer>, if Null then return <Answer>Null</Answer>.

Node level query prompt

Hello. You will be given a causal graph. The causal relationships in this causal graph are - [causal-graph-encodingbased]. Now answer using this causal graph only, is [nodeX] a [node-type] in the graph. [node-type-description]. Think step by step. Give reasoning and then give answer within <Answer> Yes/No </Answer>.

F Graphs

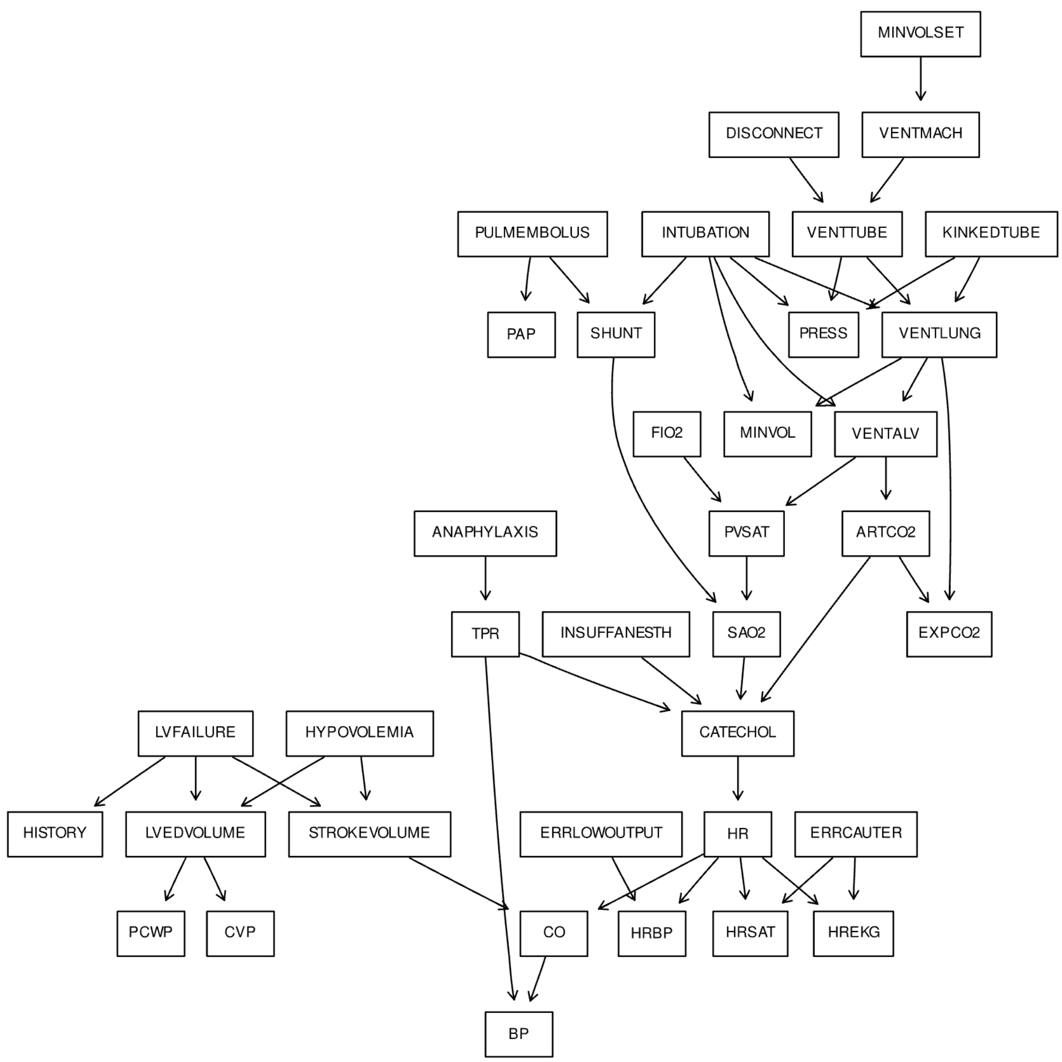

Figure 10: Alarm causal graph

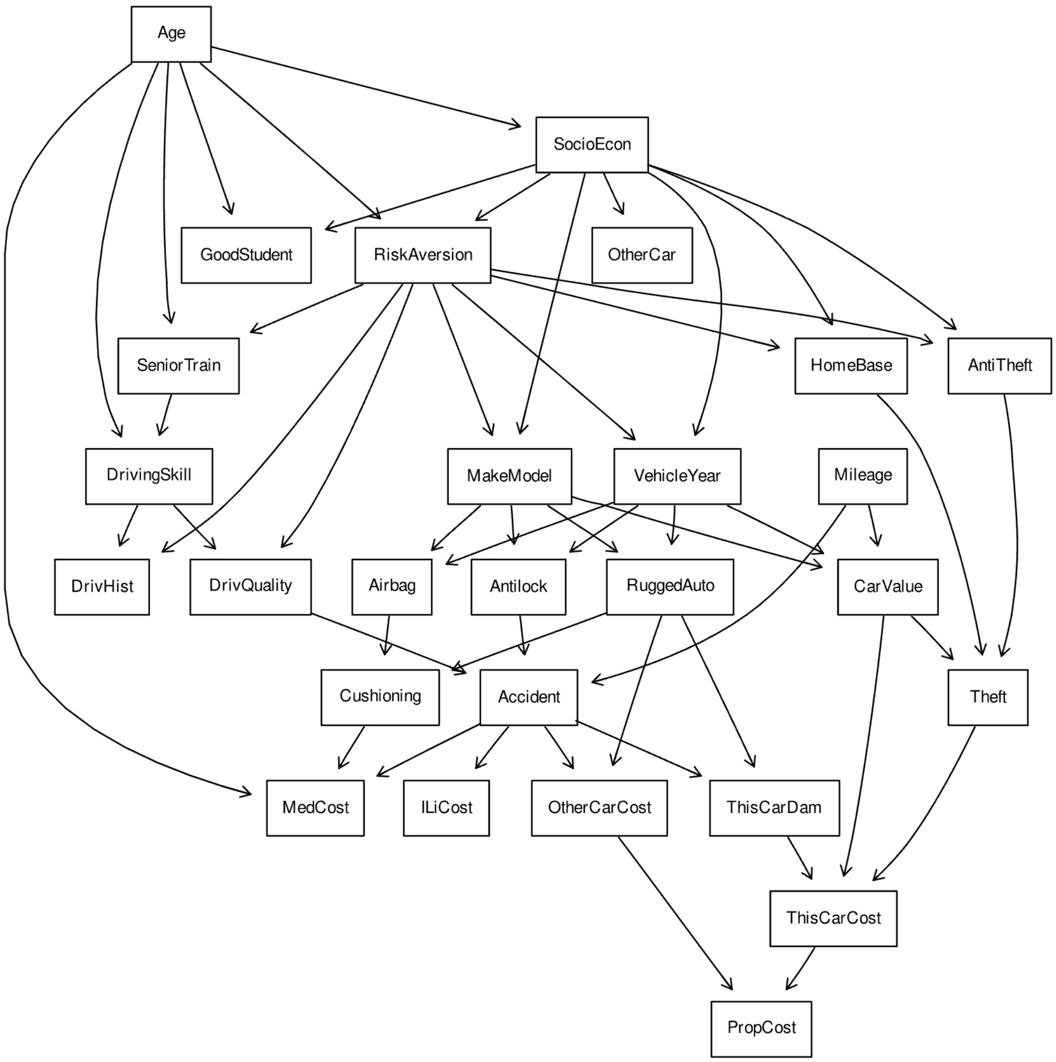

Figure 11: Insurance causal graph

G Related Works

LLMs. Instruction-tuned LLMs have become the gold standard for uncovering pre-trained knowledge via prompting [Kojima et al., 2022]. In this work, we explore the causal reasoning and causal graph understanding abilities of LLMs through prompting. LLMs have demonstrated numerous emerging abilities in language generation and certain reasoning tasks which has motivated their applications in scientific discovery [AI4Science and Quantum, 2023, Long et al., 2023, Cui et al., 2023, Demszky et al., 2023].

Causality and LLMs. Causal discovery and inference have predominantly been dominated by data-driven methods. However, due to the complexity of inferring causal structures, previous works have introduced priors on causal graphs in terms of interventions, domain expertise, edge existence, or ancestral constraints [Constantinou et al., 2023, Ban et al., 2023b, Brouillard et al., 2020]. These priors help to reduce the search spaces of potential causal graphs. Recent advancements in LLMs have motivated the use of LLM-based priors and causal discovery [Long et al., 2023, Cai et al., 2023, Anonymous, 2023, Jin et al., 2023a, Kıcıman et al., 2023]. Unlike data-driven methods, LLMs leverage causal variable names to evaluate the existence of edges between them, thereby constructing causal graphs. The rich pretrained knowledge of LLMs has proven to be almost as effective in discovering causal structures as traditional data-driven methods [Vashishtha et al., 2023, Kıcıman et al., 2023]. These initial results have motivated the integration of LLMs as priors combined with different statistical causal discovery methods. For instance, Vashishtha et al. [2023] used pairwise queries to discover the existence of edges between different causal variables and then applied methods such as PC [Spirtes et al., 2001] to reorient the edges, whereas Ban et al. [2023b] utilized LLM-based priors for scoring-based discovery methods. Vashishtha et al. [2023] suggest triplet-based prompting strategies, and Jiralerspong et al. [2024] proposed reducing the prompting complexity by prompting in a depth-first search (DFS) manner. More recently, Abdulaal et al. [2024] proposed an iterative collaboration between LLMs and structural causal models, where the LLM refines the output of SCMs. Despite their success, Jin et al. [2023a] and Zečević et al. [2023] find that LLMs are not yet fully capable of understanding true causality. Combined with external tools, Jin et al. [2023b] demonstrated the use of LLMs for causal inference tasks, albeit on 3-4 node tasks. Another line of previous works [Girju et al., 2002, Hassanzadeh et al., 2020, Tan et al., 2023] explored the use of LLMs to discover potential causal structures from unstructured data.

Most of these works assume a particular prompting strategy. However, it remains unclear which strategy would be most effective. In this paper, we aim to contribute to this line of research by benchmarking a variety of LLMs on a range of tasks related to causal graphs and exploring the effectiveness of different causal graph encoders.